香港城市大學
City University of Hong Kong

Department of Electronic Engineering

EE3314

Final Report

FishIt

CAO Jie	51737593
DING Yuhong	51800052
GAO Xiongfei	51797817
HE Meng	51800899
LIU Heyang	51799104

Group 1

Abstract

Android market enjoys an increasing market share in current touchscreen mobile devices market and gains more and more popularity day by day. Among all of the applications that Google Play provides, “game” would be viewed as the largest collection, which was finally selected as our application’s type.

In this project, a relaxing fishing game named “FishIt” is developed. This is a highly user-interactive game which simulates the ocean environment and the vivid movement of creatures, providing an amazing playing experience to the target group – young generation.

Three different playing modes were designed at the very beginning. However due to the time limitation, regretfully only one mode was well developed and function-supported at current stage. In this mode, the user can play in a leveling up scheme until the terminating condition has been met – all of the rope is used up. The difficulty will be increased as the level goes up, since the length of the rope is accumulated and the user always have to come up with the most efficient way to consume the rope. To add more variation to the game, different ocean creatures were introduced such as clownfish, sea turtle and sea horse. Scoring and ranking system was also adopted using database technology. Besides, a store was provided to the users for purchasing useful tools, adding more fun to this game.

Table of Contents

1.1 Background 3
1.2 Market Research 6
1.2.1 Target User Group 7
1.2.2 Target Platforms and Device Configurations 9
2.1 User Scenarios 11
2.2 Screen and Interactivity Design 13
2.3 Functional Requirements 14
Animation ..... 错误!未定义书签。
Fish Moving ..... 错误!未定义书签。
Boundary Detection ..... 错误!未定义书签。
Game Setting ..... 错误!未定义书签。
Scoring System ..... 错误!未定义书签。
2.4 Non-Functional Requirements 19
2.4.1 Performance requirements 19
2.4.2 Security 19
2.5 Architectural Design 20
3.1 Configuration 22
3.2 Known Issues 22
3.3 User Manual 23
3.3.1 The Welcome Page 23
3.3.2 The About Page 24
3.3.3 Game Mode 24
3.3.4 Tools 26
Fish Food 27
Water Spin 28
Freezer 28
3.3.5 Special Creatures 29
Turtle 29
Seahorse 30
3.3.6 Performance Evaluation Report 31
3.3.7 The Rank Page 33
3.3.8 The Shopping System 33
3.4 Future Development 35
4.1 Tasks Division and Role Assignment 38
4.2 Roadmap 40
5.1 Evaluation 41
References 42

List of Figures

Figure 1 Smartphone Manufacturer Share by Operating System [2] 3
Figure 2 Number of Available Applications in March 2011[3] 4
Figure 3 Google Play App Installs by September 2012 [4] 4
Figure 4 Worldwide Sessions per Category, January – February 2012 [5] 5
Figure 5 Percentage of App Downloaders Who Would Pay for an App by Category in Quarter 2, 2011 [6] 5
Figure 6 i Fishing [7] 6
Figure 7 Ninja Fishing [8] 7
Figure 8 Fishing Joy [9] 7
Figure 9 Smartphone OS Penetration by Age Group during August – October 2011 [10] 8
Figure 10 Mobile Application Reach by Age by September 2011 [11] 8
Figure 11 Top Android Apps by July 2011 [12] 9
Figure 12 Android Platform Version Distributions 10
Figure 13 Creature Design 13
Figure 14 Achieve Stable FPS 14
Figure 15 Fish It – Icon 23
Figure 16 The Welcome Page 23
Figure 17 The About Page 24
Figure 18 Fishing Method 25
Figure 19 Tools 26
Figure 20 Fish Food 27
Figure 21 Fish Food – Effects 27
Figure 22 Water Spin 28
Figure 23 Freezer 28
Figure 24 Freezer -- Effects 29
Figure 25 Turtle 29
Figure 26 Turtle for Extra money 30
Figure 27 Seahorse 30
Figure 28 Seahorse -- Effects 31
Figure 29 Performance Evaluation Report -- Stage Completion 31
Figure 30 Performance Evaluation Report -- Game Over 32
Figure 31 The Rank Page 33
Figure 32 The Shopping System 34
Figure 33 Level Selection Page 35
Figure 34 Adventure Mode Instruction 36
Figure 35 Future Instruction 37

1.1 Background

The evolution of mobile phones indicates the improvements of material living standards. People cannot live without mobile phones, which are not only just phones, but also as powerful as small computers. According to Gartner's report, over 153 million mobile devices have been sold in Quarter 2, 2012 [1]. As smartphones are becoming people's first choice when purchasing mobile phones, manufacturers like Apple, Samsung, HTC, Motorola, and Nokia are fighting vigorously for market share. Different platforms, such as IOS, Android, BlackBerry and Windows, provide products with different user experience. Among all the mobile operating systems, Android has a soaring smartphone market share, and would remain the biggest market in the following years. Figure 1 shows the smartphone manufacturer share by operating system in Quarter 2, 2012.

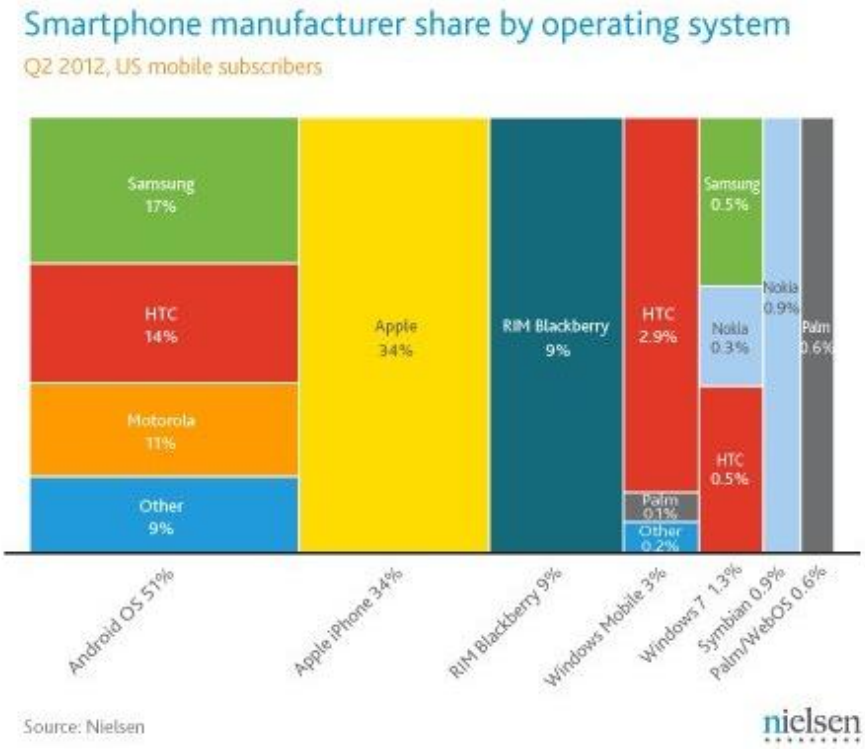


Figure 1 Smartphone Manufacturer Share by Operating System [2]

Wide price-range, touch screen, pleasant appearance, fast Internet access and various functional applications make these Android smartphones popular among all ages and incomes. The diversity of Android applications is no doubt the most notable feature

considered by smartphone purchasers. From Figure 2, we can find that Google Play (Android app store) had the second largest number of available applications among all the application stores in March 2011. What's more, as announced by Google in September 2012 (Figure 3), 25 billion apps had been downloaded, which was a figure Apple reached six months before. It looks like that Google Play still has room to grow, so that it is worthwhile spending our efforts on Android application development.

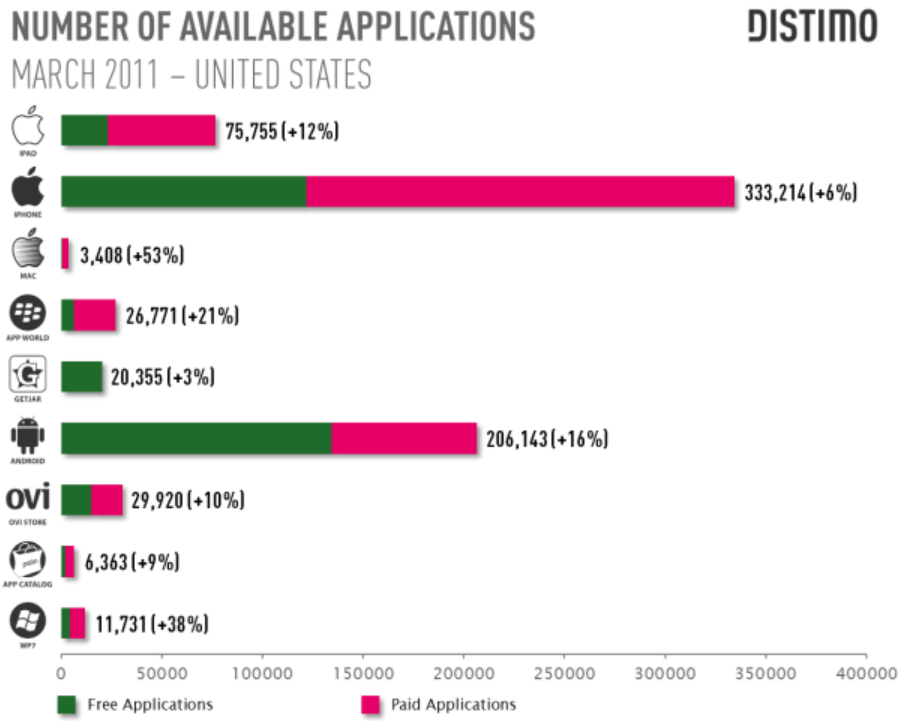


Figure 2 Number of Available Applications in March 2011[3]

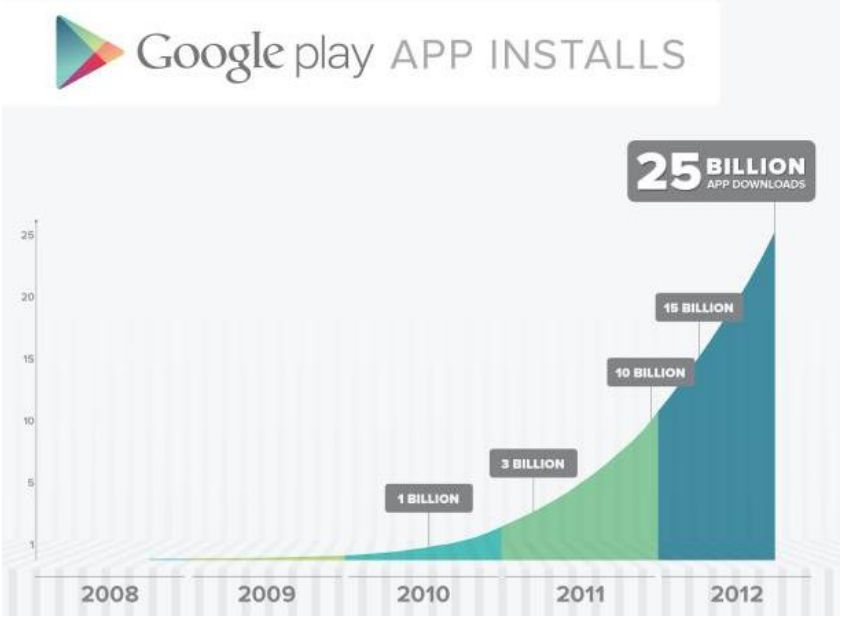


Figure 3 Google Play App Installs by September 2012 [4]

For the first two months of 2012, Flurry Analytics measured that more than half of all end user sessions were spent in games (Figure 4). Another Nielsen's Research done in Quarter 2, 2011 showed that 93% of app downloaders were willing to pay for games (Figure 5). As a result, our attempt to develop a game app will be a wise choice proved by historical statistics.

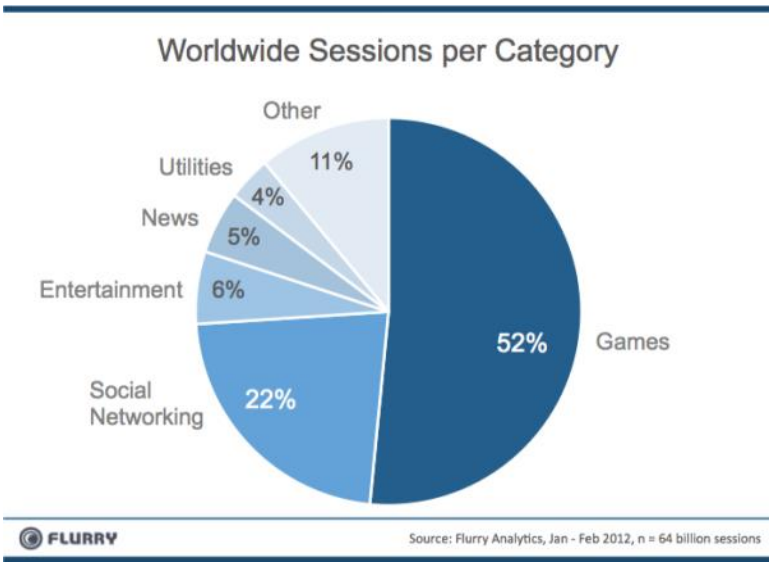


Figure 4 Worldwide Sessions per Category, January – February 2012 [5]

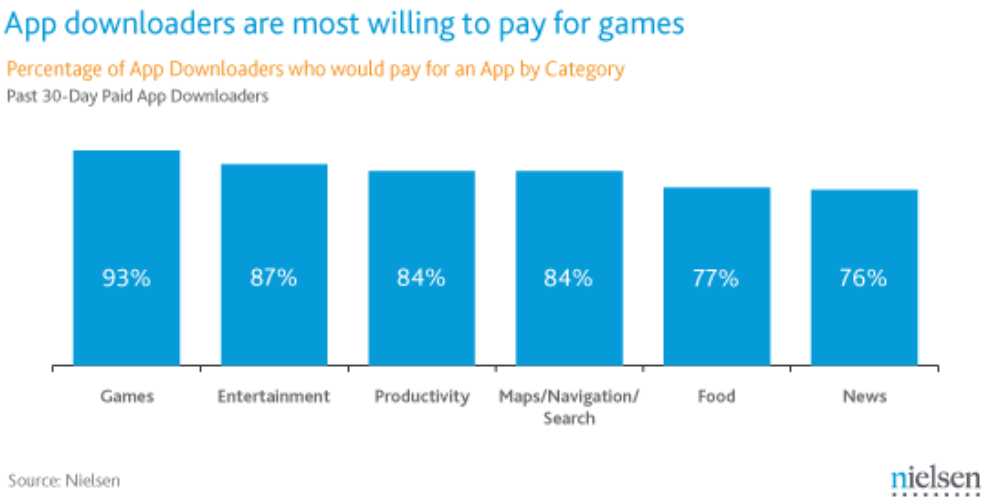


Figure 5 Percentage of App Downloaders Who Would Pay for an App by Category in Quarter 2, 2011 [6]

1.2 Market Research

In Google Play, there are currently many kinds of fishing games, which are utilizing different methods to play games. For example, i Fishing (Figure 6) simulates real fishing scene and users can drag the fishing pole to catch fish. Another fishing game is Ninja Fishing (Figure 7), which is similar to Fruit Ninja, and you can slice your fingers on the screen to cut the fish into two pieces. There is also a fishing game called Fishing Joy (Figure 8). When you tap on the screen, a piece of fishing net will appear, and fish covered by the net will be caught. There are still other fishing games, but these three are the most typical ones with relative more advantages. They all have user-friendly interfaces and good user experience with rating about 4.5 out of 5.0. Each of them has unique fishing technique, and our FishIt also has unique techniques different from all of the techniques in the market.

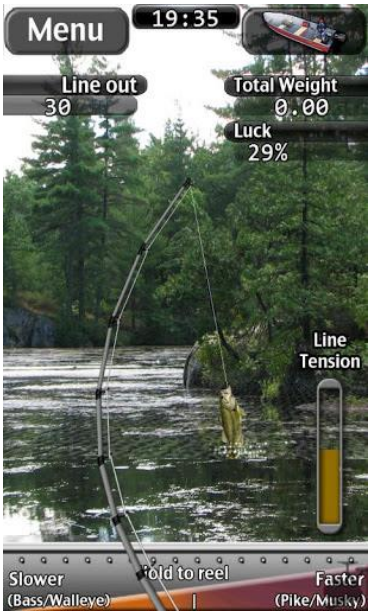


Figure 6 i Fishing [7]

Figure 7 Ninja Fishing [8]

Figure 8 Fishing Joy [9]

1.2.1 Target User Group

Age: No limitations

According to the Nielsen's research (Figure 9), Android attracts not only young consumers, but also old people above 55. Another research also showed surprising statistics (Figure 10), games like Angry Birds, appeal more to those 35-45 when compared to other age groups: 35 percent of them have used the app in the last 30 days, while only 22 percent of those 18-24 and 29 percent of 25-34 year-olds launched the game. Therefore, we can conclude that games can be ubiquitous among all age groups, and our game FishIt do not have limitations on age. As long as you can use your fingers, you can play this fishing game.

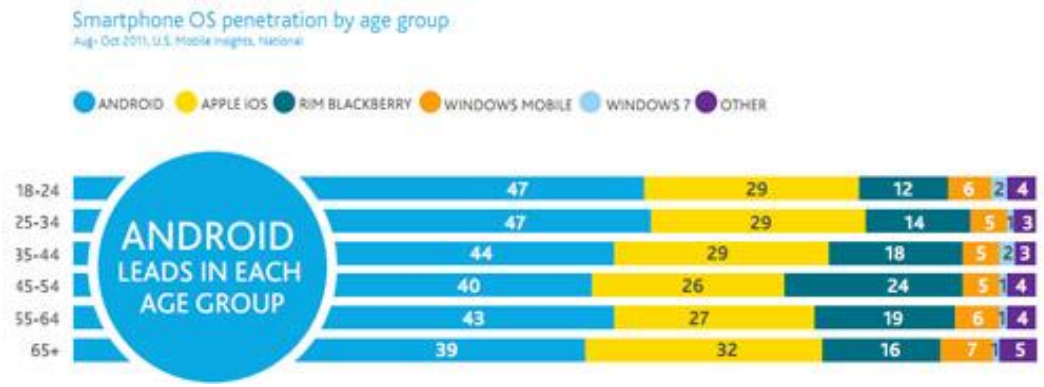

Smartphone OS penetration by age group
ANDROID
APPLE iOS
RIM BLACKBERRY
WINDOWS MOBILE
WINDOWS 7
OTHER
ANDROID LEADS IN EACH AGE GROUP
18-24 47 29 12 6 2 4
25-34 47 29 14 5 1 3
35-44 44 29 18 5 2 3
45-54 40 26 24 5 1 4
55-64 43 27 19 6 1 4
65+ 39 32 16 7 1 5

Figure 9 Smartphone OS Penetration by Age Group during August – October 2011 [10]

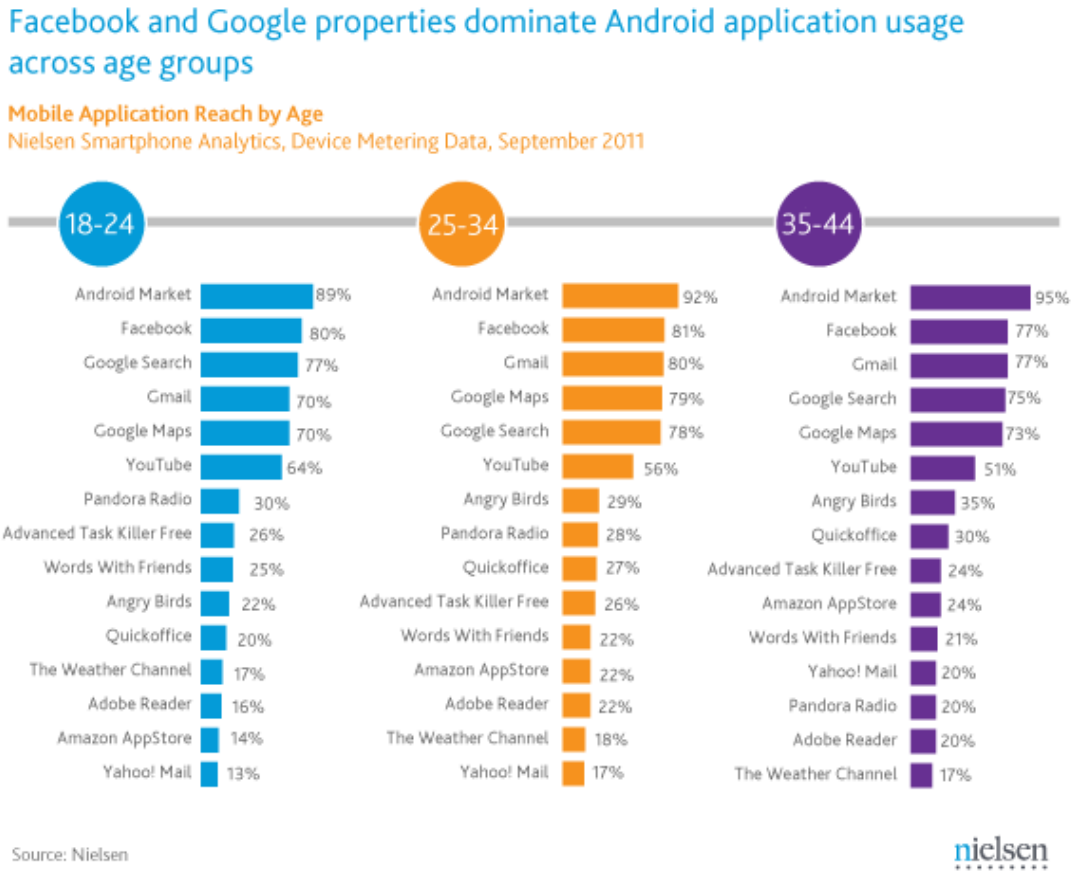


Figure 10 Mobile Application Reach by Age by September 2011 [11]

Gender: No Limitations or Preferences

According to Nielsen's research, there was little influence of gender on the choices of game apps such as Angry Birds. Therefore, FishIt will have no limitations or preferences over gender, and everyone would enjoy this kind of fishing game.

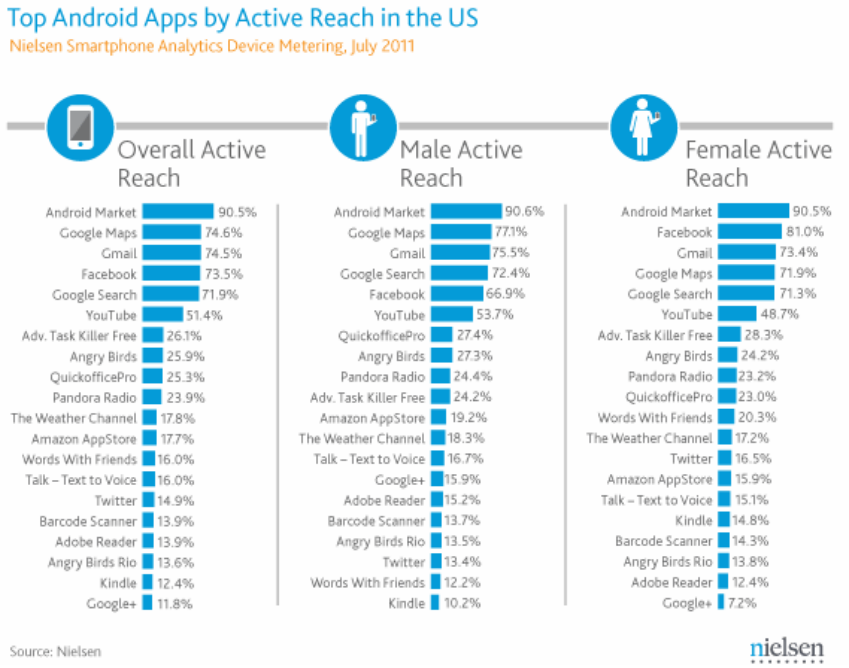


Figure 11 Top Android Apps by July 2011 [12]

1.2.2 Target Platforms and Device Configurations

The target platform for FishIt is Android 2.3 or above. The following table and chart are based on the number of Android devices that have accessed Android Market within a 14-day period ending on December 3 2012. The most popular platforms are Android 2.3.3 and 4.0.3. In order to fulfill the most customer requirements, we are going to make our application compatible with Android 2.3 or above.

Version	Codename	API	Distribution
1.5	Cupcake	3	0.10%
1.6	Donut	4	0.30%
2.1	Eclair	7	2.70%
2.2	Froyo	8	10.30%
2.3 – 2.3.2	Gingerbread	9	0.20%
2.3.3 – 2.3.7		10	50.60%
3.1	Honeycomb	12	0.40%
3.2		13	1.20%
4.0.3 – 4.0.4	Ice Cream Sandwich	15	27.50%
4.1	Jelly Bean	16	5.90%
4.2		17	0.80%

Table 1 Android Platform Version Distribution [13]

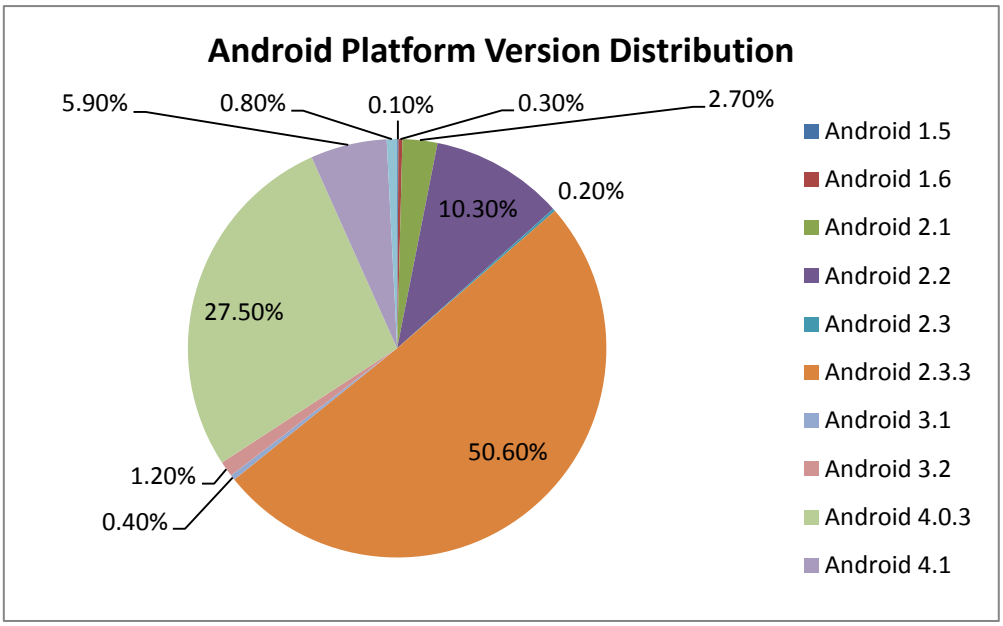


Figure 12 Android Platform Version Distributions

The device configurations are listed as follows:

- Testing Device: Samsung Galaxy Tab 8.9 LTE
- Android Version: 2.3 or above
- Processor: 1.5GHz dual-core Samsung Exynos processor
- Screen Size: 5.3 inches or above
    - As FishIt needs precise operations, we suggest users to play it on large screens.
- Screen Resolution: 1,280 x 800 pixel (WXGA) resolution
- Screen Orientation: Portrait
- Touch Screen: Full touch bar

2.1 User Scenarios

“FishIt” is a highly interactive strategy-forming game, designed to stimulate really fishing environment which targets for fishing enthusiasts. The game aims to entertain user with high-quality forged ocean setting, challenging fishing tasks, creative fishing methods, attractive user interface, and etc.

The overall design philosophy is “self-challenging” which makes our game endless. Basically, user needs to capture as many fish as possible with limited rope provided and within the limited time. The amount of time and rope, as two resources to continue the game, can be gained through satisfactory performance or can be purchased via the shopping system. With the help of various tools and special creatures, the turtle and seahorse which will be introduced detailed in the “User Manual”, user should try their utmost to maximize the number of fish caught, thus gain more coins. The game will never end until one of the two resources, i.e. rope and time, is used up. This will lead to “game over” and give out the performance evaluation report to show performance. The game keeps a recording of SIX best players in terms of their username and final score in the ranking page. Since we are planning to connect the game to social networks, like Facebook or twitter, allowing the competition with friends, the ranking system is the key.

Although the game is endless, it is still divided into certain stage, i.e. levels, to show users their detailed performance after each stage via the performance evaluation report. Base on the report, corresponding amount of the rope and coins will be awarded. And the game will automatically continue after certain delay for users to view their performance.

The “self-challenging” design makes the game even more interesting, thus attract users by arousing their passions to challenge themselves. For “Fish It” there is no

restriction on target user's age, gender, educational level, etc., therefore rather than from different purposes of using this application, the interactions that we have foreseen are mainly from the perspective of the users' operations and our expected input from them.

2.2 Screen and Interactivity Design

There are 5 main pages in the game. They are welcome page, about page, ranking page, shop page and game page. The 5 pages will be introduced in detail in 3.3 User Manual. All icons and creatures are designed by our group by using the Photoshop. We aim to create a relaxed environment with attractive, vivid overall design philosophy.

Figure 13 Creature Design

2.3 Functional Requirements

2.3.1.Animation

In this game, we designed a draw thread to achieve the animation effect. Instead of using the available methods offered by android sdk, we used thread to update the canvas. The reason is that thread offers better stability and solves ANR (Application Not Responding) problem. We set the fps to 50 in order to provide a fluid experience for users. The idea of achieving a stable fps is show below.

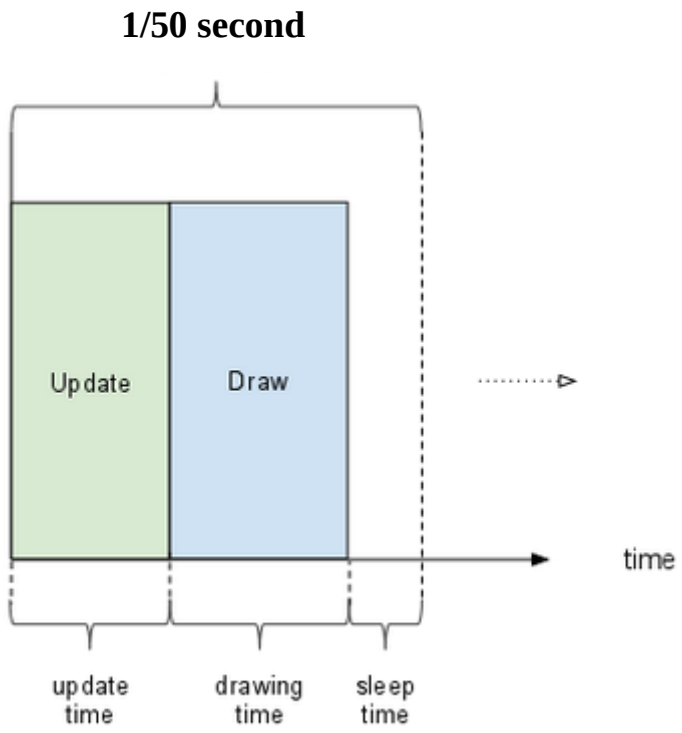


Figure 14 Achieve Stable FPS

If the sum of update time and drawing time is less than 1/50 second, then the thread will be sent to sleep for a short period. It is very useful for battery saving, too.

Pseudo code:

```
While(running){
    Catch the canvas;
    Update game state;
```

Render game state to the screen

Draw the canvas on the panel

Calculate how long did the cycle take

Calculate the sleep time

}

If sleep time > 0

Send the thread to sleep

In order to save the resource, we calculate the fps by finding the total processing time of 20 frames. We used System.nanotime() instead of System.currentTimeMillis() when calculating the fps. Because System.nanoTime() is more reliable for timing. It returns a value that is derived from some internal CPU counter/clock. The time measured by this clock cannot be changed by any user or program. System.currentTimeMillis() returns is obtained from the internal clock of the machine. If a system administrator changes the time, for example if the clock is found to be running 5 minutes fast and the administrator goes and corrects it, System.currentTimeMillis() will be affected.

```
fps = Math.round(100*1000000000.0/span*20)*0.01;
    //span is the time interval of 20 frames
```

2.3.2.Fish Moving

Currently, the fish has four moving modes. They are line, curve, searching food, and freezing. In the future, when new tools are introduced, there will be more modes.

In line mode, the position of the fish is updated according to the former position and settled angel.

```
xx = (lastX + distance * Math.cos(angle));
yy = (lastY + distance * Math.sin(angle));
```

In curve mode, the angle will be changed within a small range (-1.8° to 1.8°). When angel is larger than 360 degree,

this.angle = (this.angle) % (Math.PI * 2);

A fish's moving mode is keep changing between line mode and curve mode. Firstly, we generate a short time interval, in this time interval, the fish will move in line mode or curve mode. After that, the mode will be changed to the other one.

Pseudo code:

```
While(running){
    Generate a random time interval T
    While(T>0){
        Move in line/curve mode;
        T--;}
    If line mode
        Changing to curve mode;
        Generate a new angel randomly;
    If curve mode
        Changing to line mode;
    }
```

In the other two modes, a timer will be conducted.

When collision detected, the moving mode will be set to curve mode, and the angel will be changed within a larger range.

2.3.3.Boundary Detection

We designed a boundary detection algorithm to check whether a fish is circled or not. We used a recursive function and defined a stack in order to make the algorithm more efficient, since calling function uses system stack.

Generally, the algorithm tries to find whether there is a way from the location of fish to the boundary of screen. It makes use of a matrix of map for locations in the screen.

In this matrix, different values are set for different meanings: 1 means the location has rope, 0 means the location has no rope, and -1 means the location is not accessible.

Pseudo code:

```
public Boolean isInCircle(x, y, map){
    set current location as starting location;
    hasNext = false, a flag for whether next location is found;
    push starting location into stack;
    set starting location to -1 in map as not accessible;
    while (stack is not empty){
        if current location is screen border
            return false (fish is not inside the circle);
        hasNext = false;
        while (have not go through 4 directions && hasNext == false){
            change a direction;
            for the next location of each direction, check whether it is available for fish to go;
            if found, then hasNext = true;
        }
        if(hasNext == true){
            push the location into stack;
            set the location to -1 in map;
        }else
            pop current location out of stack;
    }
    return true (no way out, fish is inside the circle);
```

2.3.4. Database Implementation

SQLiteOpenHelper is a class that facilitates managing database creation and upgrading. By extending this class and implement its onCreate(), onUpgrad() and adding other table-wise methods to provide a complete dimension of database management functionality. The simplicity is achieved by directly implementing SQL to database of SQLiteDatabase class.
As we set the foundation for our database system, we are now ready to extend it to specific utilization. Basically, we use this class to databases for ranking database. For ranking database, we created an intermediate adapter class to provide service specific to ranking. In this class, named rankInfoService, the following methods are implemented:

```
public boolean insert(RankInfo rankinfo);
public ArrayList<RankInfo> listFirstSix();
public int getRanking(int score);
```

These three methods can facilitate update and display the ranking information flexibly, and we make use of a ArrayList for the displayed rankings, which is easy to handle and inverse the order, if necessary.

We construct a data structure for ranking information, i.e. to specify the content format to be stored into the database. Basically, there are four fields for rank entry:

```
private Integer id;
private String player;
private int score;
private Integer ranking;
```

The direct manipulations are also incorporated in the structure class.

2.3.5. Store

We use a global array to store the amount of items currently in the store. By buying items or using it, the global value will be altered and also be reflected in the store display.

2.4 Non-Functional Requirements

2.4.1 Performance requirements

Considering the requirements of maintaining our application's friendly game environment and providing enjoyable user experience, the resolution of the pictures must not be lowered to an unacceptable range, as well as the quality of the background music. Besides, the motion of the ocean creatures and many add-on features make the demands on memory even larger. After investigating the required memory of some similar ocean-environment fishing games in the current market such as Fishing Joy(58M), Ninja Fishing(43M), and Fishing Master(13M), our estimated memory will be less than 20 Megabytes. And the minimum requirement on the operating system is Android 2.2.3.

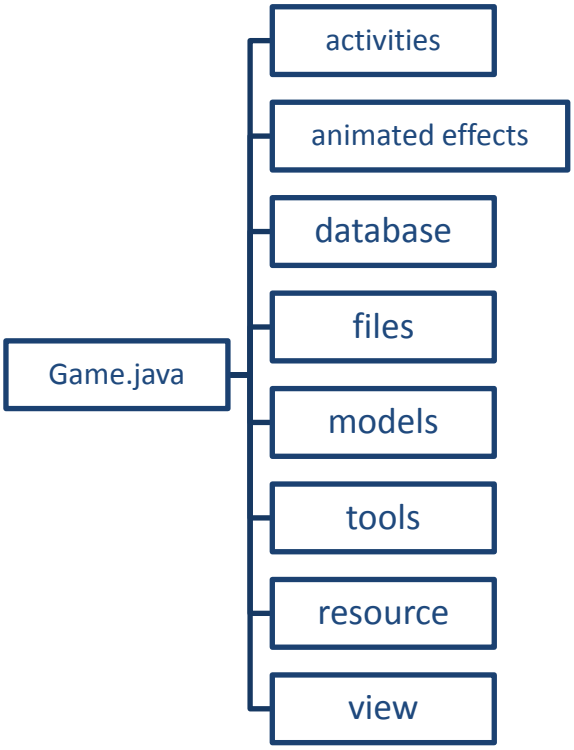
2.4.2 Security

Since in the future development stage, payment from actual currency on virtual currency in the game might be realized, the user account information is considered to be a significant security issue. We aim at achieving keeping all of the users' data secretly and it cannot be accessed by others unless being authorized.

2.5 Architectural Design

The whole project is divided into eight parts, main activity, animated effects, database, models, tools, files, resource, and view.

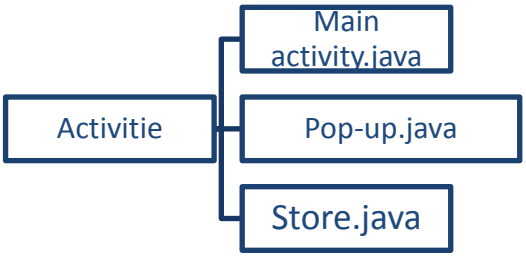
Main Structure



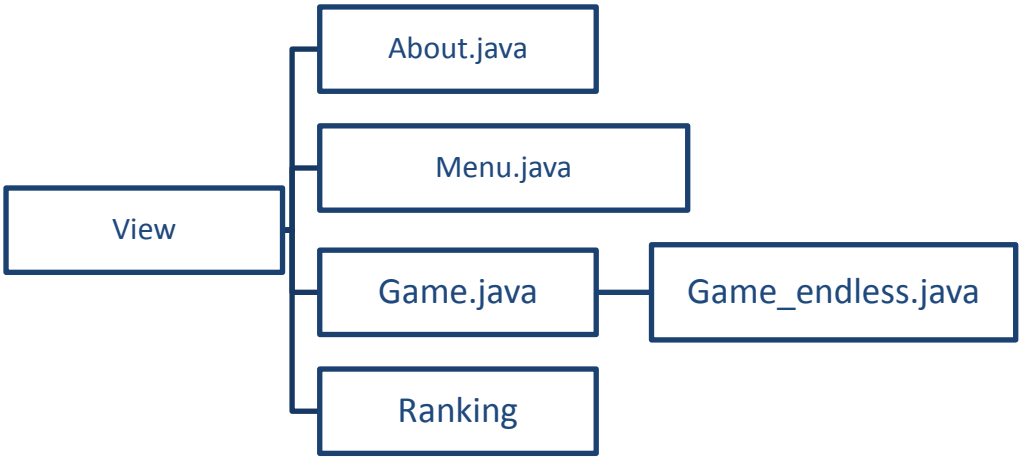
The above figure shows the structure of the main activity.

The layout of main activity is specified in main.xml. For the animation of the plant, the sequences of frames are specified in the xxx_animation.xml in the drawable folder.

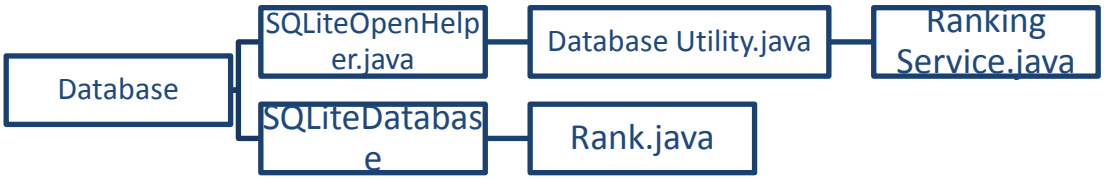
2.5.1. Activities



2.5.2. Views



2.5.1. Activities



3.1 Configuration

The application should be able to run on a touch screen supported mobile phone with the platform of Android 2.2.3 or above.

To install this application, an apk file called “FishIt” is needed which only takes up less than 10MB. At current stage, no internet connection is required.

3.2 Known Issues

As known to all, Android does not do quite well in memory management. Therefore for our application, memory crash is always one of the fetal problems which cause system error. In order not to overrun with the memory, every effort is made such as resizing the images, lowering the resolution of images as well as the bit rate of background music within acceptable range, using stack in the recursion function for detecting whether the fish is circled, and so forth.

3.3 User Manual

Figure 15 Fish It – Icon

As the above figure shows, the icon is designed with an anxiety-looking fish to link with the fishing theme. After pressing the icon, “Fish It” will be launched and the wonderful fishing time is starting!

3.3.1 The Welcome Page

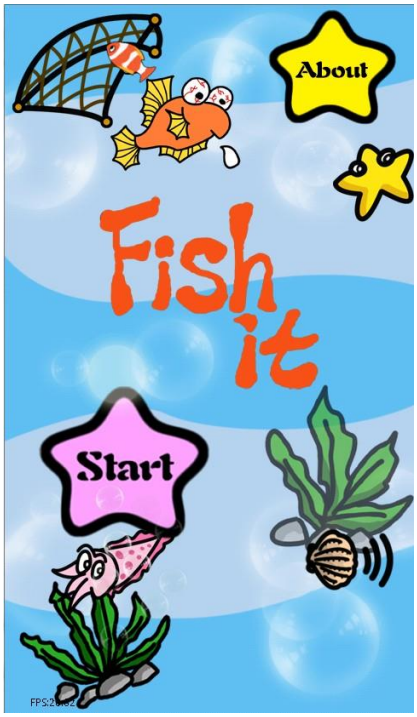


Figure 16 The Welcome Page

There are totally three functional buttons in the welcome page which will be shown

once the user launches the App. The “Start” button can simply start the game mode; the “About” button leads to the “About” page including certain information of the game; and the little shell-shaped button controls the background music and sound effects.

3.3.2 The About Page

Figure 17 The About Page

As the above screen capture shows, all the developers’ names are listed in the “About Page”. Other detailed information related to the game may be added in the later stage. The star-shaped “Back” button can lead user back to the “Welcome Page”.

3.3.3 Game Mode

As mentioned in the “User Scenarios”, user should try their utmost to catch as many fish as possible with limited amount of time and rope under the help of various tools.

One of the attractiveness of “Fish It” is its creative fishing method. In order to capture fish, user should circle the target(s) with closed shape using the rope provided from the ship located in the center of the screen.

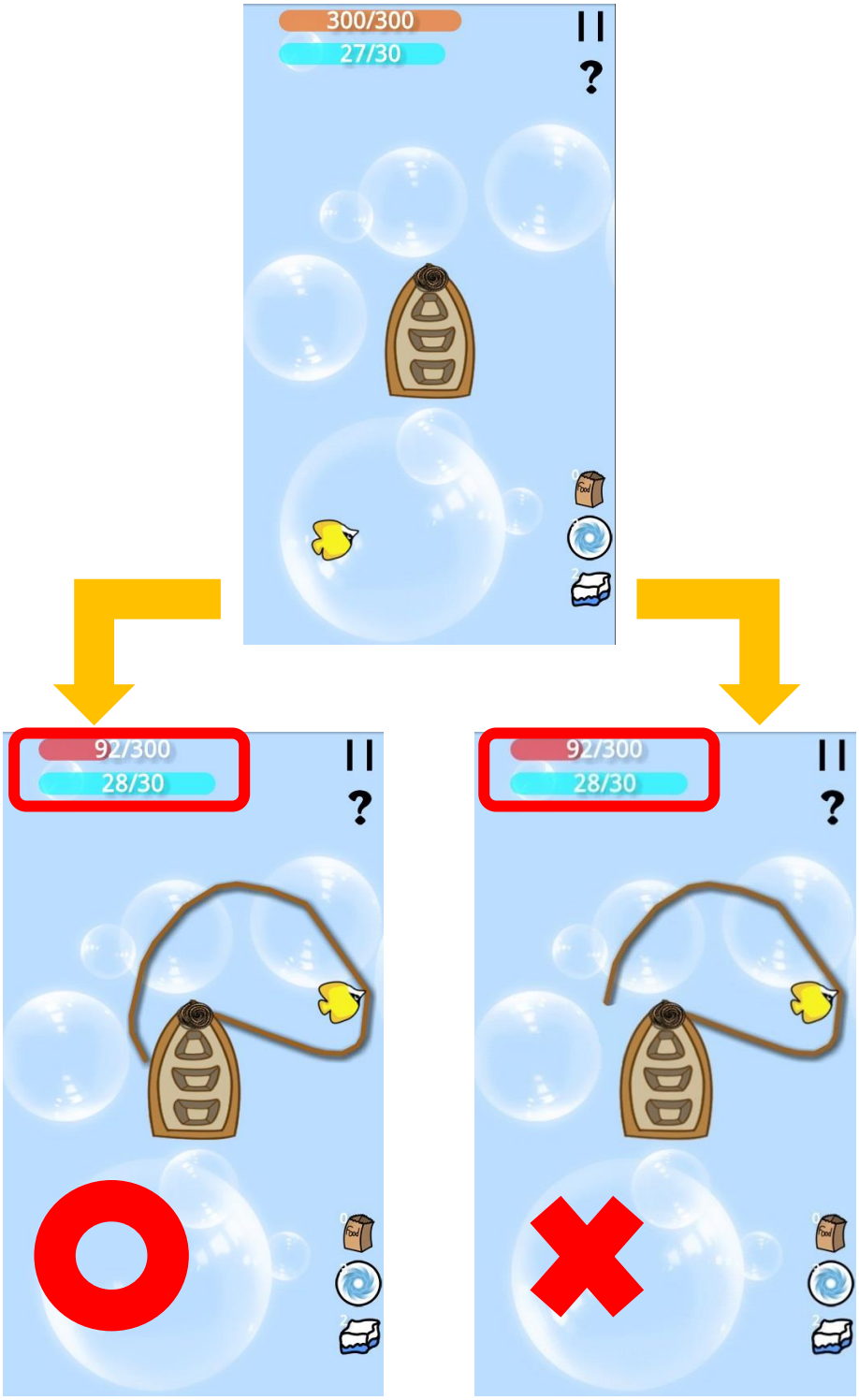


Figure 18 Fishing Method

As the above screen captures demonstrate, the rope can only be drawn from the ship or the already drawn rope. Only the closed shapes can form a compete fishing net for successfully catching the target(s).

As shown on the top of the screen, the “rope bar” and the “time bar” indicate the remaining amount of rope and time respectively.

3.3.4 Tools

“Fish It” provides user various tools to boost user performance and increase the attractiveness and interestingness to the game. There are totally 3 kinds of tools available, i.e. fish food, freezer and water spin, which can be wined through previous performance or purchased via the shopping system. However, the maximum number for each kind of tools at one time is limited to FIVE.

Figure 19 Tools

The 3 icons of the tools are listed on the lower-right corner of the screen. By touching the icon once followed by touching the place where the food tends to be placed, the corresponding tool will be placed to the desired place.

Fish Food

Figure 20 Fish Food

Fish food can attract creatures together to certain place on the screen. Usually it will be placed very close to the ship to save rope. Fish food is extremely useful for capture multiple creatures with single circle using limited amount of the rope.

Figure 21 Fish Food – Effects

Water Spin

Figure 22 Water Spin

Water Spin can force all the creatures to the button of the screen which allows easy capture.

Freezer

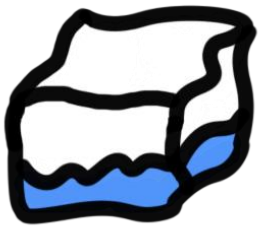


Figure 23 Freezer

Freezer, used as ice, can freeze all the creatures on the screen and make them unmoved. This largely eases the capture process.

Figure 24 Freezer -- Effects

3.3.5 Special Creatures

Besides normal fish, there are several special creatures available which will cause special effects.

Turtle

Figure 25 Turtle

Turtle represents extra money. If a turtle is captured, certain amount of money will be awarded to the user, i.e. 50 coins, as the following figure shows:

Figure 26 Turtle for Extra money

Seahorse

Figure 27 Seahorse

Seahorse represents extra rope. If a seahorse is killed by drawing a rope through it, certain amount of rope will be awarded to the user as the following figure shows:

Figure 28 Seahorse -- Effects

3.3.6 Performance Evaluation Report

The performance evaluation report will be shown under TWO situations.

One is for successfully compete the current stage. As mentioned, although the game is designed endless, still certain stages are involved.

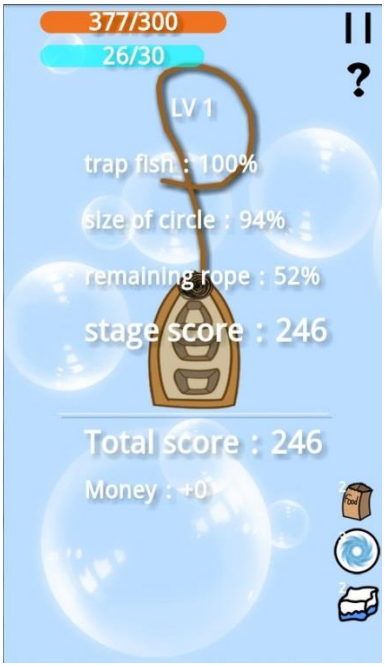


Figure 29 Performance Evaluation Report -- Stage Completion

Detailed user performance will be listed out including the percentage of fish has been caught in current stage, the size of the circle, the remaining rope and thus the total score. The report will be only shown for certain time interval and the game will continues of the report is disappeared.

The performance evaluation report can also be shown when the game is over, i.e. one of the two resources – time and rope is used up.

Figure 30 Performance Evaluation Report -- Game Over

The two reports are basically the same to include the detailed user performance, except the total number of coins owned will be list on the upper-right corner of the screen when game is over.

After game over, user can challenge again or back to the “Welcome Page”. Besides, the shopping system is now accessible for tool purchase and the ranking page is also available.

3.3.7 The Rank Page

If the user's performance is really outstanding and can refresh the records listed in the "Rank Page", the user will be asked to input name and thus become one of the SIX best player listed in the "Rank Page" with detailed usernames and scores.

Figure 31 The Rank Page

3.3.8 The Shopping System

For those useful tools as well as the amount rope and time to boost user performance, they can be purchased via store, the shopping system. Different items with required number of coins will be listed in the store. By pressing the one you like and confirm once more, the item will be gained if the remaining number of coins is sufficient.

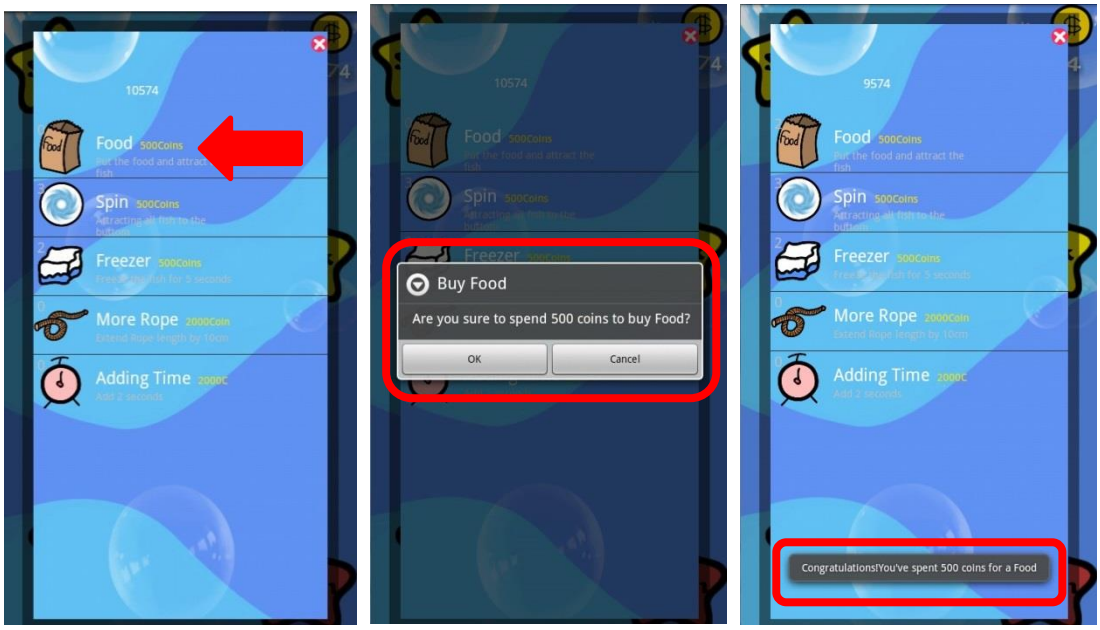


Figure 32 The Shopping System

3.4 Future Development

In the future, we will keep working on this app. We will add classical mode and adventure mode to provide users with totally different user experience. In classical mode, we will design a background story and users can enjoy the game in different scene. The “star system” will be involved. In each level user will be awarded at most three stars according to the performance.

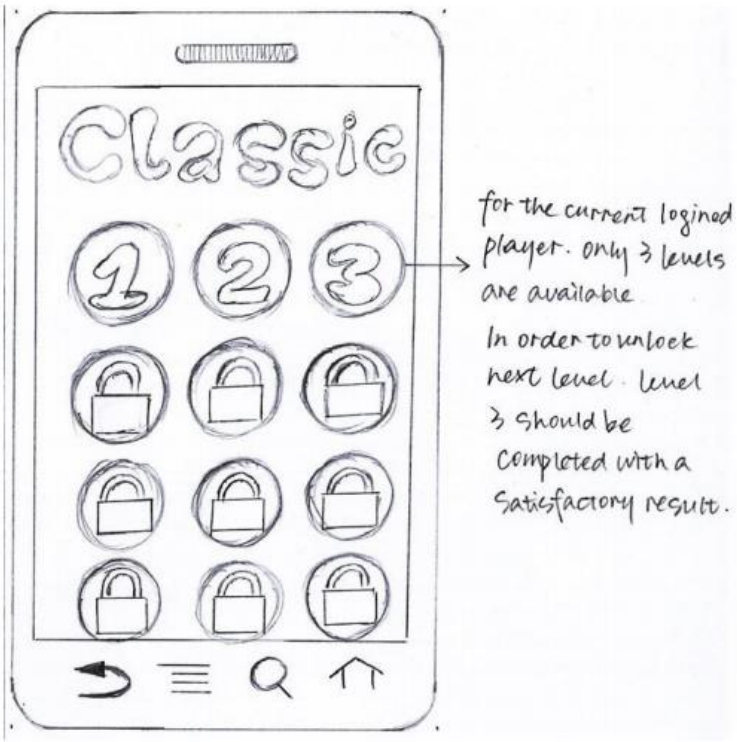

Classic
1
2
3
for the current logined player. only 3 levels are available
In order to unlock next level. level 3 should be completed with a satisfactory result.

Figure 33 Level Selection Page

In adventure mode, the actual character is presented in the figure of a diver, who is about to swim and hunt around in the deep sea to catch fish and collect treasures, both of which need good manipulation on the character’s movement. Also avoidance of fatal objects such as poisonous plants and sharks is also brought into the design. All of these scenarios are achieved by the gravity controlling sensor.

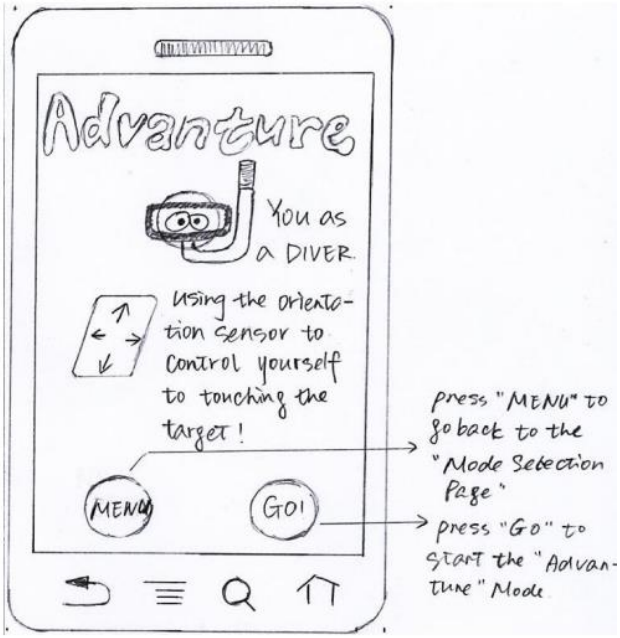

Advanture
You as a DIVER
Using the orientation sensor to control yourself to touching the target!
press "MENU" to go back to the "Mode Selection Page"
MENU
GO!
press "GO" to start the "Advanture" Mode

Figure 34 Adventure Mode Instruction

Besides new game mode, we want to offer more clear and vivid instructions via animation. We tried to insert the instructions into our game, however due to the memory limitation, the performance is poor. In the future we will find another way to achieve the instructions. After solving the memory problem, we can add more sound effects and animation effects as well.

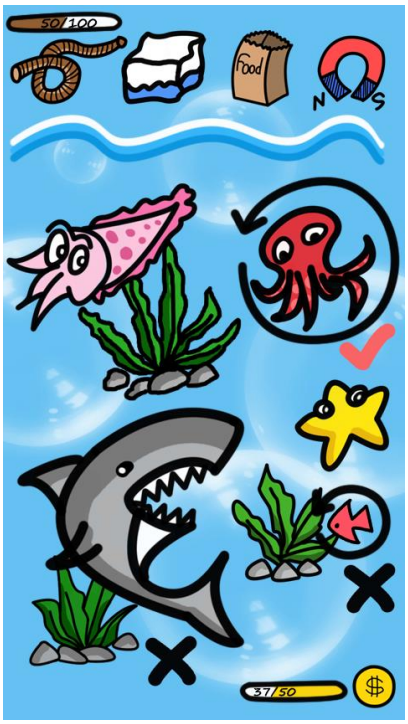


Figure 35 Future Instruction

Finally, we want to form online game circle. Users can compete with friends by sharing the score on Facebook or other social network platform. An achievement system may be involved at the same time.

4.1 Tasks Division and Role Assignment

The whole project has gone through all the following stages:

Market Research → Topic Selection → Functional Design → User Interface Design → Implementation → Documentation & Testing

The project is started by research the current Android market via secondary date and thus brain storm the topic.

After select the theme and topic, certain functionalities should be designed base on the results of market research and target customer study.

User Interface Design is further developed into THREE steps, as analysis of product and target user, visualize design using storyboard and digitalization via Photoshop.

Implementation is the main part of the development process in which the designed functionalities will be realized via coding by Java and XML. The development environment is eclipse.

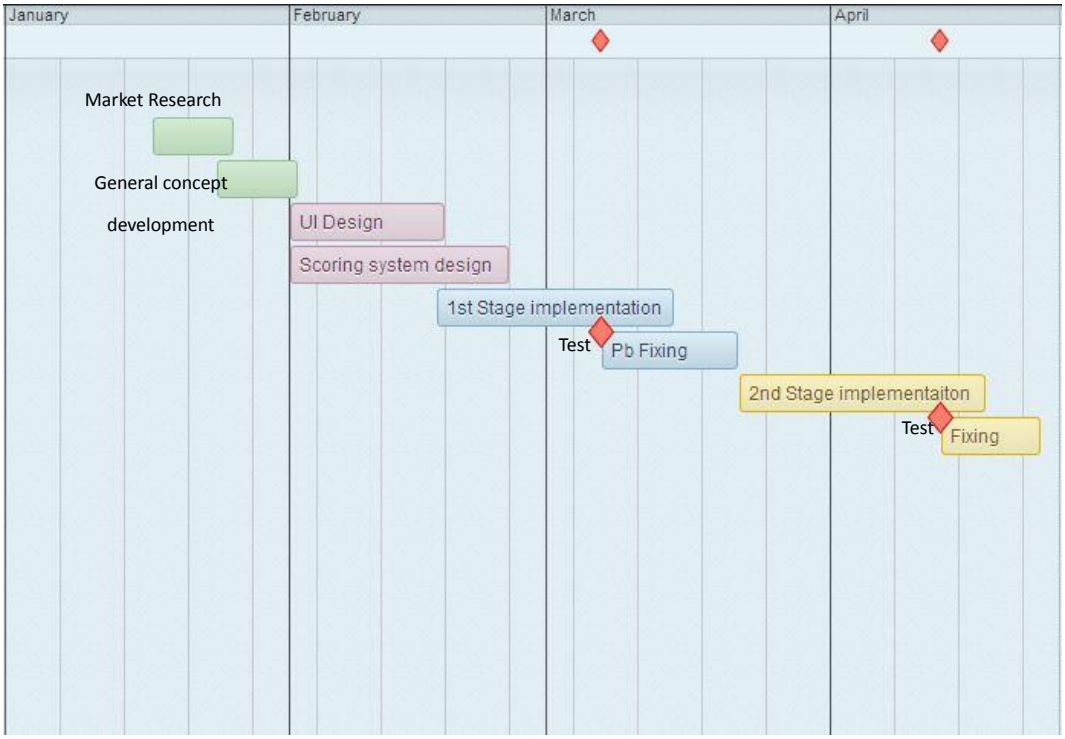
Last but not least, documentation and testing is indispensable for developing an bug-free application, at the same time, recording important things for future maintenance and user instruction.

In order to boost efficiency, we try to maximize each team member's potential by allocating the person with certain talents to do the right job. Following is the detailed role assignment table:

Name	Role	Responsibility
CAO Jie	Team Leader	● User Interface Design ● Implementation on tools ● Documentation
DING Yuhong	Team Member	● Implementation on database ● Implementation on environmental setting ● Implementation on rope & time bars
GAO Xiongfei	Team Member	● Implementation on animation ● Implementation on motion detection ● Testing
HE Meng	Team Member	● User Interface Design ● Implementation on Game Setting ● Documentation
LIU Heyang	Team Member	● Implementation on Boundary Detection ● Implementation on Scoring System ● Testing

4.2 Roadmap

The roadmap for the development of this project can be viewed as follows,



5.1 Evaluation

Although not all of the three modes were developed in the end, the final product still came out to be quite satisfactory and can be evaluated as an “entertaining game”. The User Interface is quite attractive and the proposed functionalities and features for the Survival Mode have been managed to be supported. Other than these, this application can still be further improved from the following aspects,

(1) Providing a tutorial video/animation to the users to make sure all the functions and scoring principles have been introduced clearly;

(2) Designing more useful tools to increase the complexity of this game;

(3) Designing other barriers such as “negative points gaining” objects to further increase the difficulty as level goes up;

(4) Providing online services, such as enabling the user to register and login with user account which keeps the past record for each specific user, or connect this game to Facebook, Twitter, Weibo or other social networking pages, to create an online game circle for friends and families to play and maybe compete together;

(5) More personalized sound effects can be added, such as different sound for the fish’s different status – normal, hungry, and dead.

To conclude, the ideas for designing a game can never be enough. Innovation and creativity will always be the most important thing in developing an application. Our team has demonstrated the superior quality in both of the innovation and the implementation. Every group member has contributed to the whole and with continuous cooperation and coordination, the final result appears to be satisfactory.

References

[1] http://www.gartner.com/newsroom/id/2120015

[2]http://www.engadget.com/2012/07/12/nielsen-has-android-near-52-percent-of-us-smartphone-share-in-q2/

[3]http://techcrunch.com/2011/04/27/there-are-now-more-free-apps-for-android-than-for-the-ios-platform-distimo/

[4]http://news.cnet.com/8301-1035_3-57521252-94/can-apples-app-store-maintain-its-lead-over-google-play/

[5] http://blog.flurry.com/default.aspx?Tag=Usage%20Statistics

[6]http://blog.nielsen.com/nielsenwire/online_mobile/games-most-popular-mobile-app-category/

[7]https://play.google.com/store/apps/details?id=com.RockingPocketGames.iFishing

[8]https://play.google.com/store/apps/details?id=com.gamenauts.ninjafishingfull

[9]https://play.google.com/store/apps/details?id=org.cocos2dx.FishGame

[10]http://www.electronista.com/articles/11/12/15/nielsen.q3.look.back.shows.age.splits/

[11]http://thenextweb.com/google/2011/12/12/nielsen-reveals-most-popular-android-apps-by-age-angry-birds-appeals-most-to-over-35s/

[12]http://www.intomobile.com/2011/09/14/top-20-android-apps-by-gender-led-by-facebook-google-maps-and-gmail/

[13] http://www.apps4android.org/?p=4168

[14]http://www.javacodegeeks.com/2011/07/android-game-development-game-loop.html

[15] http://www.xuanyusong.com/archives/category/android/android_game